M183 Technische Dokumentation

# Berechtigungskonzept

Das ERP, welches ursprünglich für das Modul 226b erstellt, und nun ebenfalls die Implementation des Moduls 118 verwendet wurde, hat an sich kein effektives Berechtigungskonzept, sprich Seiten oder ähnliches, auf welche nur gewisse Benutzergruppen Zugriff haben sollten. Dennoch habe ich dem Usermodel ein Rollenfeld zugewiesen, welches für diesen Anwendungsfall gedacht wäre. Das Rollenfeld ist ein Textfeld.

Damit dennoch etwas in diesem Rahmen implementiert ist, habe ich auf der Übersichtsseite, der Seite, auf welche man nach der TOTP-Validierung gelangt, einen Button implementiert, welcher nur sichtbar ist, falls dem User die entsprechende «special»-Rolle zugeteilt wurde. Diese Rolle kann ein Benutzer nur erhalten, indem man die Rolle direkt auf der Datenbank manuell bearbeiten geht. Falls man auf die «geheime» Seite gelangen möchte, kann man dies via localhost:8080/special tun.

# Gesamtstruktur

Das Projekt, in welches die Implementation für das Modul 183 implementiert wurde, war zuvor grundsätzlich eine simple, ERP-inspirierte, CRUD Applikation, welche ein einfaches Login besass.

# Eingesetzte Mittel

Die Applikation verwendet JavaScript und Embedded JavaScript als Programmiersprache. Als Framework wird Express.js verwendet. Die Datenverwaltung wird auf einer MongoDB übernommen, bzw. einem MongoDB-Cluster. Das Cluster ermöglicht es, dass die Datenbank jederzeit verfügbar ist und die Applikation stets verwendet werden kann bzw. sollte.

# Logisches Datenmodell

Die MongoDB ist eine non-relationale Datenbank, daher gibt es keine erzwungenen ForeignKey-Beziehungen, wie man es von SQL-Datenbanken kennt. Dennoch sind vom Code her einige Beziehungen herstellbar. Daher habe ich keine direkte Verbindung unter den Datenmodellen gezogen. Dennoch habe ich markiert, welche Felder FK darstellen sollten. Diese werden jedoch NIRGENDS validiert oder Ähnliches.

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

# Dokumentation relevanter Prozesse und Abläufe

Der Loginprozess beginnt auf der Seite der Registrierung (register.ejs). Von da aus kann man seine credentials eingeben. Dabei wird auf das Passwort geprüft, ob dies verschiedene Anforderungen erfüllt (14 Zeichen, min. 1 Grossbuchstaben, min. 1 Kleinbuchstaben, min. 1 Nummer, min. 1 Sonderzeichen (z.B. \_, -, / etc.).

Werden diese Anforderungen erfüllt, gelangt man auf eine weitere Seite, welche ein QR-Code darstellt. Scannt man diesen mit einer Authenticator-App, wird von da an das entsprechende TOTP-Token angezeigt, welche für das Login unerlässlich ist.

Das Scannen des QR-Codes liegt in der Verantwortung des Nutzers. Sobald dieser weiterklickt, ist der QR-Code nicht mehr aufrufbar zum jetzigen Zeitpunkt.

Nach dem Scannen gelangt man zurück zur Startseite, von dieser aus navigiert man sich weiter zur Loginseite.

Auf der Loginseite ist es benötigt sich erst mit Benutzernamen und Passwort einzuloggen. Wird kein Benutzer mit den eingegebenen Credentials gefunden, wird man wieder auf die Loginseite geleitet, gemeinsam mit einer Meldung, dass die eingegebenen Credentials nicht vorhanden sind und man erneut versuchen muss sich einzuloggen.

Wurde der Benutzername mit dem dazugehörigen Passwort gefunden, gelangt man weite zu einer Seite mit einem Feld für die Eingabe des TOTP-Tokens.

Wenn man auf dieser Seite ein falsches Token eingibt, ladet die Seite neu und es wird eine Fehlermeldung angezeigt, welche den Nutzer darauf hinweist, dass das Token ungültig ist. Ist das Token jedoch valid, wird man eingeloggt und gelangt zur Startseite.

*Für visuelle Darstellung siehe nächste Seite.*

